In the Specification:

The paragraph beginning on page 4, line 8 is amended as follows:

FIG. 10 depicts FIG. 1 to illustrate a communication link between a second web server of the web server cluster and a first application server of the application server cluster and a communication link between the first application server of the application server cluster and a first database server of the database server cluster, in accordance with embodiments of the present invention.

The paragraph beginning on page 6, line 4 is amended as follows:

Definitionally, a web server is software that serves files in response to requests from web browsers (or other requesting software). When the web server receives a request for a static HTML (i.e., HyperText Markup Language) page, the server reads the request, finds the page, and sends it to the requesting browser (or other requesting software). Common web servers include Microsoft IIS, Netscape Enterprise ~~Server~~ Server®, Java WebServer, and Apache HTTP Server. The web server may communicate with database servers to access the information needed to be served to the requesting web browser. Such communication between a web server and a database server is direct when a static HTML page is requested by the web browser, as exemplified in FIGS. 5-7 which will be discussed *infra*.

The paragraph beginning on page 11, line 15 is amended as follows:

Although the preceding functionality of the load balancers L_W, L_A, and L_D is beneficial, said load balancers do not facilitate optimizing the processing of the entire application because

the load balancer and servers of a given cluster are not apprised of non-functioning servers of other clusters. Accordingly, the load balancer and servers of the given cluster cannot adjust their processing of applications in a way that takes account the non-operational status of one or more servers in another cluster. For example, if it were possible for the load balancer and servers of the web server cluster 21 to ~~learns~~ learn that one or more application servers of application cluster 22 have become non-operational, then the web sever cluster 21 would be able to reduce the number of concurrent requests made for the services of application cluster 22 and instead divert some or all of its applications to another fully operational application server cluster until such time that all clusters of application cluster 22 have become fully operational. Unfortunately, current load balancers do not have knowledge that one or more servers of another cluster have become non-operational. The control server 25 of the present invention solves this problem by communicating with the servers and load balancers of the system 20, as will be explained *infra* in conjunction with FIGS. 16-19.

The paragraph beginning on page 14, line 15 is amended as follows:

As an example of how service nodes may be utilized, consider a situation in which the control server 25 has determined that application server A_1 of the application server cluster 22 is non-operational. Then the control server 25 may communicate with the service node S_A over the communication channel 51 to direct the ~~communication channel 51~~ service node S_A to make a determination of a cause of the application server A_1 being non-operational. Upon making said determination, the service node S_A may facilitate making the application server A_1 operational (e.g., by fixing the problem associated with the cause of the application server A_1 being non-

operational). The service node S_A may utilize the communication paths 52 and/or 53 to help determine the cause of the application server A_1 being non-operational and/or facilitate making the application server A_1 operational.